

Figure 1: Examples of notes with identical MIDI note numbers being spelt differently in different tonal contexts (from p. 8 of Piston, W. (1978). *Harmony*. Victor Gollancz Ltd., London. Revised and expanded by Mark DeVoto).



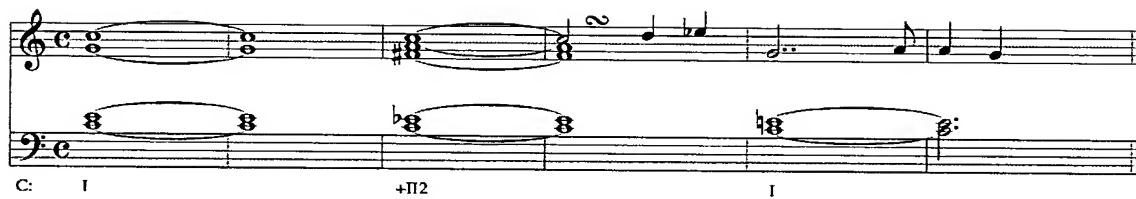


Figure 2: Should the Ebs be spelt as D#s? (From p. 390 of Piston, W. (1978). *Harmony*. Victor Gollancz Ltd., London. Revised and expanded by Mark DeVoto).





(a) 

(b) { $\langle 2, D4, 2 \rangle, \langle 4, E4, 2 \rangle, \langle 6, F4, 2 \rangle, \langle 8, G4, 2 \rangle, \langle 10, E4, 2 \rangle, \langle 12, F4, 1 \rangle,$
 $\langle 13, D4, 1 \rangle, \langle 14, CS4, 1 \rangle, \langle 15, D4, 1 \rangle, \langle 16, BF4, 4 \rangle, \langle 20, G4, 4 \rangle, \langle 24, A4, 5 \rangle,$
 $\langle 29, G4, 1 \rangle, \langle 30, F4, 1 \rangle, \langle 31, E4, 1 \rangle, \langle 32, G4, 1 \rangle, \langle 33, F4, 1 \rangle, \langle 34, E4, 1 \rangle,$
 $\langle 35, D4, 1 \rangle, \langle 36, E4, 2 \rangle, \langle 38, C5, 3 \rangle, \langle 41, B4, 1 \rangle, \langle 42, A4, 1 \rangle, \langle 43, B4, 1 \rangle,$
 $\langle 44, B4, 1 \rangle, \langle 45, A4, 1 \rangle, \langle 46, GS4, 1 \rangle, \langle 47, A4, 1 \rangle, \langle 26, A3, 2 \rangle, \langle 28, B3, 2 \rangle,$
 $\langle 30, C4, 2 \rangle, \langle 32, D4, 2 \rangle, \langle 34, B3, 2 \rangle, \langle 36, C4, 1 \rangle, \langle 37, A3, 1 \rangle, \langle 38, GS3, 1 \rangle,$
 $\langle 39, A3, 1 \rangle, \langle 40, F4, 4 \rangle, \langle 44, D4, 4 \rangle$ }

Figure 3: (a) Bars 1 to 4 of Bach's Fugue in D minor from Book 1 of *Das Wohltemperirte Klavier* (BWV 851). (b) The *OPND* representation of the score in (a).





	...	C \flat	G \flat	D \flat	A \flat	E \flat	B \flat	F	C	G	D	A	E	B	F \sharp	C \sharp	G \sharp	D \sharp	A \sharp	E \sharp	...
<i>Sharpness</i>	...	-7	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7	8	9	10	11	...

Figure 4: The line of fifths





Figure 5: The first step in Cambouropoulos's method involves converting the input representation into a sequence of pitch classes. The music is processed a window at a time. Each window contains 9 notes and the first third of each window overlaps the last third of the previous window

8



```

1   $S \leftarrow \langle \rangle$ 
2   $Q \leftarrow \langle 0, 1, 1, 2, 2, 3, 3, 4, 5, 5, 6, 6 \rangle$ 
3  for  $c \leftarrow 0$  to 11
4       $m_0 \leftarrow 0$ 
5       $s \leftarrow \langle \rangle$ 
6       $s' \leftarrow \langle \rangle$ 
7      for  $i \leftarrow 0$  to  $|C| - 1$ 
8           $s \leftarrow s \oplus \langle Q[(C[i] - c) \bmod 12] \rangle$ 
9           $m_0 \leftarrow s[0]$ 
10     for  $i \leftarrow 0$  to  $|C| - 1$ 
11          $s' \leftarrow s' \oplus \langle (s[i] - m_0) \bmod 7 \rangle$ 
12      $S \leftarrow S \oplus \langle s' \rangle$ 
13 return  $S$ 

```

Figure 6: Algorithm for computing the spelling table S .



```

1   $R \leftarrow \langle \rangle$ 
2  for  $i \leftarrow 0$  to  $|C| - 1$ 
3       $V \leftarrow \langle 0, 0, 0, 0, 0, 0, 0 \rangle$ 
4       $\mu \leftarrow \langle \rangle$ 
5      for  $k \leftarrow 0$  to 11
6           $\mu \leftarrow \mu \oplus \langle S[k][i] \rangle$ 
7      for  $j \leftarrow 0$  to 11
8           $s_{\text{prev}} \leftarrow V[\mu[j]]$ 
9           $V[\mu[j]] \leftarrow s_{\text{prev}} + \Phi[i][j]$ 
10      $R \leftarrow R \oplus \langle POS(\max(V), V) \rangle$ 
11 return  $R$ 

```

Figure 7: Algorithm for computing the relative morph list R .





```
1   $H \leftarrow \langle\langle O[0] \rangle\rangle$ 
2  for  $i \leftarrow 1$  to  $|C| - 1$ 
3      if  $O[i][0] = O[i-1][0]$ 
4           $H[|H| - 1] \leftarrow H[|H| - 1] \oplus \langle O[i] \rangle$ 
5      else
6           $H \leftarrow H \oplus \langle\langle O[i] \rangle\rangle$ 
7  return  $H$ 
```

Figure 8: Algorithm for computing the chord list H .





Figure 9: Neighbour note and passing note errors corrected by *ps13*.





```

1  for  $i \leftarrow 0$  to  $|H| - 3$ 
2     $\zeta \leftarrow \langle \rangle$ 
3    for  $k \leftarrow 0$  to  $|H[i + 2]| - 1$ 
4       $\zeta \leftarrow \zeta \oplus \langle H[i + 2][k][1, 3] \rangle$ 
5    for  $n_1 \leftarrow 0$  to  $|H[i]| - 1$ 
6      if  $H[i][n_1][1, 3] \in \zeta$ 
7        for  $n_2 \leftarrow 0$  to  $|H[i + 1]| - 1$ 
8          if  $H[i + 1][n_2][2] = H[i][n_1][2]$ 
9            if  $(H[i + 1][n_2][1] - H[i][n_1][1]) \bmod 12 \in \{1, 2\}$ 
10              $H[i + 1][n_2][2] \leftarrow (H[i + 1][n_2][2] + 1) \bmod 7$ 
11            if  $(H[i][n_1][1] - H[i + 1][n_2][1]) \bmod 12 \in \{1, 2\}$ 
12              $H[i + 1][n_2][2] \leftarrow (H[i + 1][n_2][2] - 1) \bmod 7$ 
13  return  $H$ 

```

Figure 10: Algorithm for correcting neighbour note errors.





```

1  for  $i \leftarrow 0$  to  $|H| - 3$ 
2    for  $n_1 \leftarrow 0$  to  $|H[i]| - 1$ 
3      for  $n_3 \leftarrow 0$  to  $|H[i+2]| - 1$ 
4        if  $H[i+2][n_3][2] = (H[i][n_1][2] - 2) \bmod 7$ 
5          for  $n_2 \leftarrow 0$  to  $|H[i+1]| - 1$ 
6            if  $H[i+1][n_2][2] = H[i][n_1][2]$  or  $H[i+1][n_2][2] = H[i+2][n_3][2]$ 
7              if  $0 < (H[i][n_1][1] - H[i+1][n_2][1]) \bmod 12 < (H[i][n_1][1] - H[i+2][n_3][1]) \bmod 12$ 
8                 $\zeta \leftarrow \langle \rangle$ 
9                for  $j \leftarrow 0$  to  $|H[i+1]| - 1$ 
10                  if  $H[i+1][j][2] = (H[i][n_1][2] - 1) \bmod 7$ 
11                     $\zeta \leftarrow \zeta \oplus \langle H[i+1][j] \rangle$ 
12                 $\theta \leftarrow \langle \rangle$ 
13                for  $j \leftarrow 0$  to  $|\zeta| - 1$ 
14                  if  $H[i+1][n_2][1] \neq \zeta[j][1]$ 
15                     $\theta \leftarrow \theta \oplus \langle \zeta[j] \rangle$ 
16                if  $\theta = \langle \rangle$ 
17                   $H[i+1][n_2][2] \leftarrow (H[i][n_1][2] - 1) \bmod 7$ 
18 return  $H$ 

```

Figure 11: Algorithm for correcting descending passing note errors.





```

1  for  $i \leftarrow 0$  to  $|H| - 3$ 
2    for  $n_1 \leftarrow 0$  to  $|H[i]| - 1$ 
3      for  $n_3 \leftarrow 0$  to  $|H[i+2]| - 1$ 
4        if  $H[i+2][n_3][2] = (H[i][n_1][2] + 2) \bmod 7$ 
5          for  $n_2 \leftarrow 0$  to  $|H[i+1]| - 1$ 
6            if  $H[i+1][n_2][2] = H[i][n_1][2]$  or  $H[i+1][n_2][2] = H[i+2][n_3][2]$ 
7              if  $0 < (H[i+2][n_3][1] - H[i+1][n_2][1]) \bmod 12 < (H[i+2][n_3][1] - H[i][n_1][1]) \bmod 12$ 
8                 $\zeta \leftarrow \langle \rangle$ 
9                for  $j \leftarrow 0$  to  $|H[i+1]| - 1$ 
10                  if  $H[i+1][j][2] = (H[i][n_1][2] + 1) \bmod 7$ 
11                     $\zeta \leftarrow \zeta \oplus \langle H[i+1][j] \rangle$ 
12                 $\theta \leftarrow \langle \rangle$ 
13                for  $j \leftarrow 0$  to  $|\zeta| - 1$ 
14                  if  $H[i+1][n_2][1] \neq \zeta[j][1]$ 
15                     $\theta \leftarrow \theta \oplus \langle \zeta[j] \rangle$ 
16                if  $\theta = \langle \rangle$ 
17                   $H[i+1][n_2][2] \leftarrow (H[i][n_1][2] + 1) \bmod 7$ 
18 return  $H$ 

```

Figure 12: Algorithm for correcting ascending passing note errors.





```
1  $O' \leftarrow \langle \rangle$ 
2 for  $i \leftarrow 0$  to  $|H| - 1$ 
3    $O' \leftarrow O' \oplus H[i]$ 
4  $M' \leftarrow \langle \rangle$ 
5 for  $i \leftarrow 0$  to  $|O'| - 1$ 
6    $M' \leftarrow M' \oplus \langle O'[i][2] \rangle$ 
7 return  $M'$ 
```

Figure 13: Algorithm for computing M' .





```

1   $P \leftarrow \langle \rangle$ 
2  for  $i \leftarrow 0$  to  $|J| - 1$ 
3     $o_1 \leftarrow \lfloor J[i][1]/12 \rfloor$ 
4     $o_2 \leftarrow 1 + o_1$ 
5     $o_3 \leftarrow o_1 - 1$ 
6     $p_1 \leftarrow o_1 + M'[i]/7$ 
7     $p_2 \leftarrow o_2 + M'[i]/7$ 
8     $p_3 \leftarrow o_3 + M'[i]/7$ 
9     $c \leftarrow J[i][1] \bmod 12$ 
10    $p' \leftarrow o_1 + c/12$ 
11    $D \leftarrow \langle |p' - p_1|, |p' - p_2|, |p' - p_3| \rangle$ 
12    $\omega \leftarrow \langle o_1, o_2, o_3 \rangle$ 
13    $o \leftarrow \omega[POS(\min(D), D)]$ 
14    $P \leftarrow P \oplus \langle M'[i] + 7o \rangle$ 
15 return  $P$ 

```

Figure 14: Algorithm for computing P .





```

PPN( $p$ )
1   $m \leftarrow p[1] \bmod 7$ 
2   $L \leftarrow \langle \text{"A"}, \text{"B"}, \text{"C"}, \text{"D"}, \text{"E"}, \text{"F"}, \text{"G"} \rangle$ 
3   $l \leftarrow L[m]$ 
4   $g_c \leftarrow p[0] - 12(\lfloor p[1]/7 \rfloor)$ 
5   $A \leftarrow \langle 0, 2, 3, 5, 7, 8, 10 \rangle$ 
6   $c' \leftarrow A[m]$ 
7   $e \leftarrow g_c - c'$ 
8   $i \leftarrow \text{" "}$ 
9  if  $e < 0$ 
10   for  $j \leftarrow 0$  to  $-e - 1$ 
11      $i \leftarrow i \oplus \text{"f"}$ 
12 else
13   if  $e > 0$ 
14     for  $j \leftarrow 0$  to  $e - 1$ 
15        $i \leftarrow i \oplus \text{"s"}$ 
16   else
17      $i \leftarrow \text{"n"}$ 
18  $o_m \leftarrow \lfloor p[1]/7 \rfloor$ 
19 if  $m = 0$  or  $m = 1$ 
20    $o \leftarrow o_m$ 
21 else
22    $o \leftarrow 1 + o_m$ 
23  $o_{\text{str}} \leftarrow STR(o)$ 
24 return  $l \oplus i \oplus o_{\text{str}}$ 

```

Figure 15: PPN algorithm.

